ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΕΠΛ231: Δομές Δεδομένων και Αλγόριθμοι

Εαρινό Εξάμηνο 2013

Φροντιστήριο 2 - ΛΥΣΕΙΣ

Άσκηση 1

Να αναλύσετε τον χρόνο εκτέλεσης των πιο κάτω σαν συνάρτηση του n.

A.

```
sum = 0;
for (int i = 1; i <= n²; i++)          //C
     for (j = 1; j <= 2n², j++)        //B
          for (k = 1; k <= n/2; k++) //A
               sum++;
```

$$A = \sum_{k=1}^{\frac{n}{2}} 1 = \frac{n}{2}$$

$$B = \sum_{j=1}^{2n^2} \frac{n}{2} = 2n^2.\frac{n}{2} = n^3$$

$$C = \sum_{i=1}^{n^2} n^3 = n^2.n^3 = n^5 \in \Theta(n^5)$$

B.

```
sum = 0;
for (i = 1; i ≤ lg n; i++)       //B
     for (j = 1; j ≤ i²; j++)    //A
          sum++;
```

$$A = \sum_{j=1}^{i^2} 1 = i^2$$

$$B = \sum_{i=1}^{lgn} i^2 = \frac{lgn(lgn+1)(2lgn+1)}{6} \in \Theta(lg^3 n)$$

Γ.

```
sum = 0;
for (int i = 1; i <= n²; i++)          //C
     for (j = 1; j <= n; j*=2)         //B
          for (k = 1; k <= j; k++)     //A
               sum++;
```

Λύση Γ1

Παρατηρούμε ότι οι ο εσωτερικός βρόγχος εξαρτάται από το μεσαίο, αλλά η αύξηση του μεσαίου μετρητή δεν είναι γραμμική. Για αυτό τον λόγο δεν μπορούμε να χρησιμοποιήσουμε αθροίσματα για τον υπολογισμό του χρόνου εκτέλεσης (εκτός και αν μετατρέψουμε το μεσαίο μετρητή σε γραμμικό – Λύση Β), αλλά πρέπει να χρησιμοποιήσουμε πίνακα δουλεύοντας από μέσα προς τα έξω. Υποθέτουμε χωρίς βλάβη της γενικότητας ότι το n είναι δύναμη του 2.

j=	1	2	4	n/4	n/2	n
T(A,B)=	1	2	4	n/4	n/2	n

Ο χρόνος εκτέλεσης του μεσαίου βρόγχου είναι ίσος με το άθροισμα του χρόνου εκτέλεσης κάθε επανάληψης του εξωτερικού βρόχου, δηλαδή:

$T2,3(j) = 1 + 2 + \ldots + n/4 + n/2 + n = 2n - 1 \in \Theta(n)$

Ο εξωτερικός βρόγχος είναι ανεξάρτητος των εσωτερικών.

$$C = \sum_{i=1}^{n^2} n = n^3 \in \Theta(n^3)$$

Λύση Γ2

<u>Η λύση αυτή είναι εφικτή επειδή τα I, j, k δεν χρησιμοποιούνται για οποιεσδήποτε άλλες πράξεις μέσα στο πλαίσιο κώδικα.</u>

Μετατρέπουμε τον μεσαίο βρόγχο σε γραμμικό ώστε να χρησιμοποιήσουμε αθροίσματα για τον υπολογισμό του χρόνου εκτέλεσης. Με την ίδια ανάλυση της λύσης Γ1.

Θέλουμε το for (j = 1; **j<=n**; **j*=2**) να γίνει γραμμικό, δηλ., for (j = 1; **j<=X**; **j++**). Συνεπώς, για να εκτελεί ο προσαρμοσμένος βρόγχος τον ίδιο αριθμό επαναλήψεων, θα πρέπει το X να ισούται με lg n.

Αφού το j επηρεάζει το for (k = 1; k <= j; k++) τότε πρέπει να το μετατρέψουμε και αυτό ώστε να διατηρήσουμε τον ίδιο αριθμό επαναλήψεων.

Για j=	1	2	4	8	2^z
Πριν j=lg n	1	2	4	8	2^z
Μετά j=lg n	-	1	2	3	z

Συνεπώς, μετασχηματίζουμε το for (k = 1; k <= j; k++)
σε for (k = 1; k <= 2^j; k++)

Άρα

```
for (int i = 1; i <= n²; i++)          //C
     for (j = 1; j <= lg n; j++)       //B
          for (k = 1; k <= 2^j; k++)   //A
               sum++;
```

$$A = \sum_{k=1}^{2^j} 1 = 2^j$$

$$B = \sum_{j=1}^{lgn} 2^j = 2^{lgn+1} - 1 = 2n - 1 \in \Theta(n)$$

$$C = \sum_{i=1}^{n^2} n = n^3 \in \Theta(n^3)$$

Δ.

```
sum = 0;
for (i = 1; i ≤ n; i++)                //C
     for (j = 1; j ≤ n+i; j++)         //B
          for (k = j; k <= n; k++)     //A
               sum++;
```

$$A = \sum_{k=j}^{n} 1 = n - j + 1$$

$$B = \sum_{j=1}^{n+i} n - j + 1 = \sum_{j=1}^{n+i} n + 1 - \sum_{j=1}^{n+i} j = (n+1)(n+i) - \frac{(n+i)(n+i+1)}{2} = \cdots$$

$$= \frac{n^2 + n + i - i^2}{2}$$

$$C = \sum_{i=1}^{n} \frac{n^2 + n + i - i^2}{2} = \frac{1}{2}\sum_{i=1}^{n} \frac{n^2 + n + i - i^2}{2} = \cdots = \in \Theta(n^3)$$

Άσκηση 2

Έστω T(n) ο χρόνος εκτέλεσης της factorial (n). Τότε η τιμή του T(n) δίνεται από την πιο κάτω αναδρομική εξίσωση:

$T(1) = 1$ (Αν n=1 η διαδικασία κάνει δύο πράξεις, μια σύγκριση και το return:$2 \in O(1)$)

$T(n) = T(n-1) + 1$ (Αν n>1 η διαδικασία έχει χρόνο εκτέλεσης όσο η κλήση factorial(n-1), δηλαδή, σύμφωνα με τον ορισμό (*) T(n–1), και δύο επιπλέον πράξεις ένα πολλαπλασιασμό και το return)

Θα λύσουμε την αναδρομική εξίσωση με τη μέθοδο της αντικατάστασης:

$$
\begin{aligned}
T(n) &= T(n-1) + 1 \\
&= T(n-2) + 1 + 1 \\
&= T(n-2) + 2 \\
&= T(n-3) + 3 \\
&= \dots \\
&= T(n-i) + i \\
&= T(1) + (n-1) \qquad \text{//έστω } i=n-1 \text{ ώστε να επιτύχουμε } T(n-i)=T(n-n+1)=T(1) \\
&= 1 + (n-1) \\
&= n \in \Theta(n)
\end{aligned}
$$